Proceedings of the NECDC 2002

LA-MPI: The Los Alamos Message Passing Interface

David J. Daniel, Nehal N. Desai, Richard L. Graham,
L. Dean Risinger and Mitchel W. Sukalski

Advanced Computing Laboratory (CCS-1), MS B287, Los Alamos National Laboratory,
Los Alamos NM 87545, U.S.A.

We present an overview of the Los Alamos Message Passing Interface. LA-MPI is an end-to-end network-fault-tolerant message passing system targeting high performance for parallel scientific applications on terascale clusters. It is portable, open source and implements MPI v1.2, the de facto standard for parallel scientific computing.

1. Introduction

High-performance computing has traditionally been the domain of expensive, special-purpose vector and/or parallel supercomputers. Recently, however, supercomputer-level performance has become achievable using large clusters of commodity-based systems using open source software. Given the inexpensive nature of these solutions it is highly probable that increasingly large clusters will come to dominate high-performance computing.

One consequence of the rise of cluster computing is a growing concern with *fault tolerance* of communication networks. This is because the manufacturing tolerance to which commodity hardware conforms may be inadequate to guarantee error-free execution of an application, given the length of a typical application run, and the very large number of individual systems that are aggregated into a cluster. For example, a commodity network device may have an error rate which is perfectly acceptable for a desktop system, but not in a cluster of thousands of nodes which must run error free for hours or days to successfully complete a scientific calculation.

The Los Alamos Message Passing Interface, LA-MPI, has been developed to address this concern, while simultaneously providing excellent performance. LA-MPI is a standard compliant implementation of MPI v1.2, the *de facto* standard for parallel scientific computing, which guarantees data integrity while exploiting the full performance of the network device. Furthermore, LA-MPI is open source and written in a modular fashion making it readily portable to new architectures.

This paper gives a brief overview of the features, performance and fault tolerance capabilities currently provided and planned for LA-MPI.

2. Features

LA-MPI provides the following features:

- MPI v1.2 compliant
- Delivers data reliably
- Thread safe
- High performance (exploiting low-level network device capabilities)
- Portable, open source
- Simultaneous use of multiple network paths:
  - Message striping across *heterogeneous* paths (i.e. network devices of different types)
  - Message-fragment striping on *homogeneous* paths (i.e. network devices of the same type)
- Supported platforms:
  - Processor architectures including Intel x86, Alpha, MIPS, PowerPC
  - Operating systems including Linux, IRIX, Tru64, MacOS X
  - Network paths including IP (UDP), Quadrics, Myrinet, HIPPI800

3. Architecture

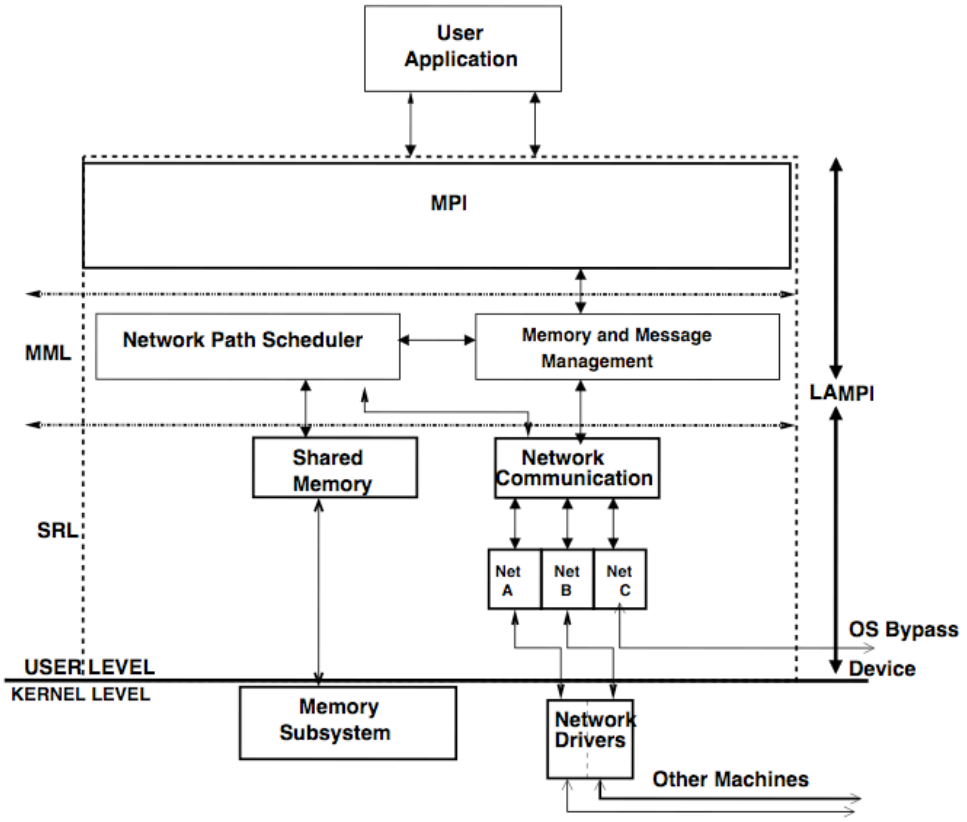

Figure 1: LA-MPI architecture

LA-MPI has a three-tier architecture, as outlined in Figure 1.

MPI Layer

This layer provides a thread-safe MPI 1.2 compliant interface for compatibility with existing applications.

Memory and Message Layer

The memory manager controls all memory, both local and remote. The path scheduler "binds" a specific message between given source and destination processes to a particular network path, an abstraction used to encapsulate the properties of network devices and protocols.

Send and Receive Layer

This layer is responsible for sending and receiving message fragments, and is highly network dependent.

4. Performance

LA-MPI is designed to be a *high-performance* message-passing library. Performance results on the Nirvana cluster at LANL (a cluster of 128-way SGI Origin 2000s networked with HIPPI800) highlight the key areas of latency, bandwidth and scalability.

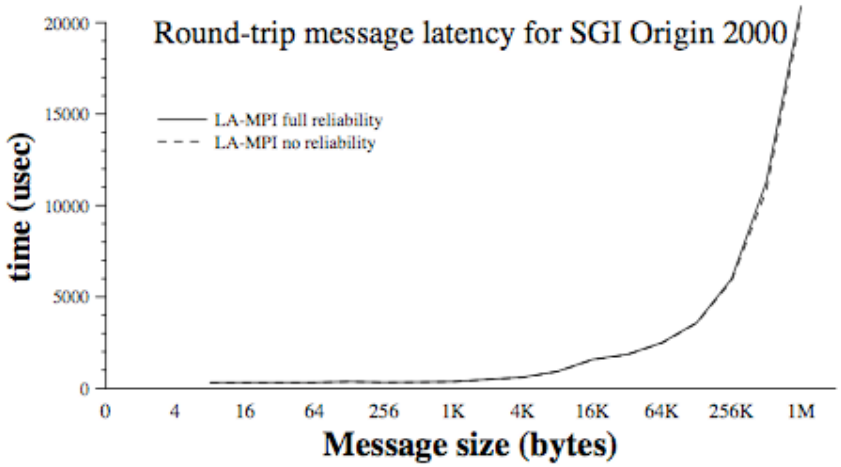

Figure 2

Figure 2 presents ping-pong latency results, showing that the overhead of implementing reliable messaging is minimal.

Bandwidth results are given in Figure 3, and equal or surpass other MPI implementations. Note the effect of message-fragment striping, utilizing multiple network paths setting in at 64 kilobytes.

MPI collective operations have been optimized for scalability. For example, Figure 4 presents results for MPI_Reduce, showing logarithmic scaling with the number of hosts in the cluster.

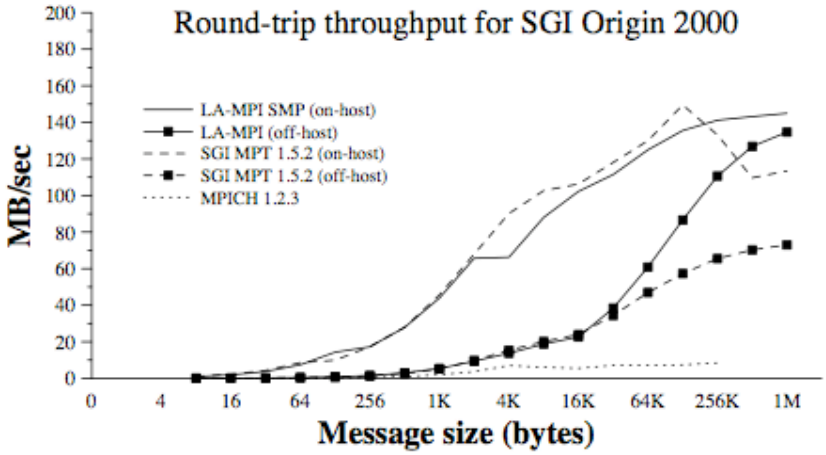

Figure 3

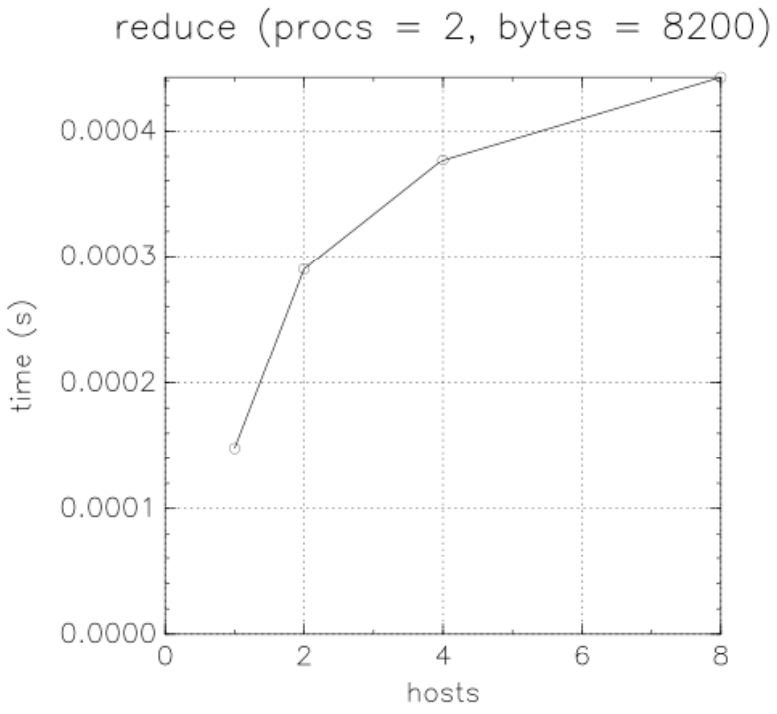

Figure 4

5. Fault Tolerance

To address the issues of fault tolerance discussed in the introduction, LA-MPI aims to be the first high-performance fault-tolerant message-passing system.

Available Now...

Network-Error Reliability. Message data integrity is guaranteed at the application level via a 32-bit checksum or CRC. If corruption due to hardware or system software is detected, data is retransmitted. This occurs transparently to the user.

Coming Soon...

- Network-Failure Resilience. If a network device fails, traffic will "fail over" to another existing path: the path scheduler will "rebind" outstanding messages to another functional route between the source and destination processes.
- "Fail back" to the first route, corresponding to the case of a temporarily unavailable network device, is also planned.
- Support for fault–tolerant algorithms, by allowing applications to take decisions on whether to continue after a process failure (behavior undefined in MPI).
- Optimizations for scalable start-up.
- Dynamic process management including:
    - Process migration
    - MPI 2 support (as needed by customers)

Research areas...

- Application-level fault tolerance is addressable once dynamic process management is in place. Automatic and incremental check-pointing schemes are being investigated.
- Integration with other HPC Linux thrusts, including the Clustermatic/Bproc Single System Image project that includes predictive system-health monitoring.

References

Graham, R.L., Choi, S.E., Daniel, D.J, Desai, N.N, Minnich R.G., Rasmussen C.E., Risinger, L.D., Sukalski, M.W. "A Network-Failure-Tolerant Message-Passing System for Terascale Clusters," Proceedings of the International Conference on Supercomputing ICS'02, New York, NY, 2002.